

Carlos Vido

Aluno do [curso de Ciências Moleculares](#), desenvolvendo projeto de pesquisa em modelagem de proteínas com foco em drug design no [LnBio](#).

Meus Exercícios

[1ex.r](#) [2ex.r](#) [3ex.r](#) [4ex.r](#) [5ex.r](#) [6ex.r](#) [7ex.r](#) [8ex.r](#) [9ex.r](#)

Proposta de Trabalho Final

Principal

Implementação de uma rotina que, tendo como entrada um arquivo no formato FASTA¹⁾, estima o pI²⁾ da proteína ou peptídeo correspondente e sua carga em um determinado pH. Os cálculos serão feitos utilizando dados de pK³⁾ encontrados na literatura.

A entrada é o nome do arquivo que contém a sequência, um parâmetro opcional que indica o banco de dados de onde se deseja retirar os pKs (a ser escolhido dentre 6 opções, o default sendo a Wikipedia) e outro que indica o pH a partir do qual calcular a carga da proteína (entre 0 e 14, o default sendo 7.0). O pI será calculado da seguinte maneira:

$$pI = \text{mean}(pK)$$

O resultado será utilizado para estimar se a proteína tem carga positiva ou negativa, permitindo o uso da equação de Henderson-Hasselbalch mais adequada a cada caso: se o pH da solução é maior que o pI da proteína,

$$\text{charge} = \sum(-1/(1 + 10^{(pK_n - pH)}))$$

onde pK_n são os pKs dos aminoácidos com cadeia lateral carregada negativamente. Se o pI é maior que o pH da solução,

$$\text{charge} = \sum(1/(1 + 10^{(pH - pK_n)}))$$

considerando desta vez apenas os aminoácidos com cadeia lateral com carga positiva. Finalmente, por definição, se pH = pI,

$$\text{charge} = 0$$

A saída da função será um data frame contendo o nome do arquivo de entrada, o pI calculado, o nome do banco de dados escolhido pelo usuário, a carga calculada e o pH relativo a este cálculo.

Comentários

Daniel:

Não tenho o conhecimento necessário para compreender o que sua função se dispõe a fazer. Você poderia detalhar melhor? Quais seriam as manipulações feita no objeto gerado pelo arquivo FASTA? Qual seria a formato de saída (retorno ao usuário) de sua função?

Ale: Também me falta embasamento. Sei que tem o conhecimento necessário para faze-lo, mas seria bom que na documentação e mesmo na página aqui tentasse explicar em termos menos exclusivos da área... Foi muito telegráfico. Vale também, como para a equação de Hendersson-Hasselbach, colocar os links para que possamos entender os termos que não fazem parte do nosso universo(PI da proteína, FASTA, pK).

Paulo: já conversamos pessoalmente, vai fundo!

Kadu:

Dei uma expandida. Tinha deixado curtinho porque não sabia se ia ser aceito, me digam se está melhor agora.

Plano B

Aceito sugestões.

Página de Ajuda

PACOTE DE ANÁLISE ELÉTRICA BÁSICA DE PROTEÍNAS

diagnostics.r

Contém:

```
luciferase read.fasta(filename, header=TRUE) protein.charge(protein, pk = wiki, ph = 7.0)  
protein.diagnostics(filename, header=TRUE, db = "wk", ph = 7.0)
```

:: luciferase O arquivo sem extensão "luciferase" contém a proteína homônima expressa no formato FASTA, retirada do site do NCBI para permitir a demonstração do pacote.

Exemplo de uso:

```
system("cat luciferase") ## OBS: o comando pressupõe o uso de um sistema baseado no Unix.
```

```
:: read.fasta(filename, header=TRUE)
```

A função 'read.fasta' abre um arquivo de nome 'filename', que pode ou não ter um cabeçalho (de uma única linha) a ser ignorado pela função. O parâmetro lógico 'header' (com default 'TRUE') indica se a primeira linha deve ser descartada.

O arquivo é transformado em um fator contendo cada aminoácido individualmente, devolvido ao usuário. Isso é feito para facilitar a análise eletroquímica da molécula. NÃO É FEITA consistência de dados para verificar se o arquivo não pertencia originalmente a uma molécula de DNA.

Um 'warning' avisa se há aminoácidos desconhecidos no arquivo.

Exemplos de uso:

```
source("protein.diagnostics.r") luci ← read.fasta("luciferase") summary(luci)

:: protein.charge(protein, pk = wiki, ph = 7.0)
```

A função 'protein.charge' abre um fator contendo uma proteína e utiliza uma manipulação da equação de Hendersson-Hasselbalch para calcular a carga da proteína em um determinado pH. O parâmetro 'pk' recebe um vetor numérico com nomes contendo o valor de pK a ser considerado para os aminoácidos carregados e para os terminais da proteína. Valores retirados da Wikipedia constituem o default. O parâmetro 'ph', com default 7,0, indica o pH em que o usuário deseja que o cálculo de carga seja realizado.

Exemplo de uso:

```
source("protein.diagnostics.r") luci ← read.fasta("luciferase") protein.charge(luci, header=T, ph = 10.0)

:: protein.diagnostics(filename, header=TRUE, db = "wk", ph = 7.0)
```

A função 'protein.diagnostics' utiliza as duas outras funções do pacote para abrir um arquivo de proteína sob o nome 'filename', salvar seu conteúdo como um fator intitulado 'protein' criado fora do escopo da função e calcular parâmetros eletroquímicos referentes à proteína. Estes incluem carga a um determinado pH, número de aminoácidos com carga positiva e negativa e pl.

Para os cálculos, a função contém uma tabela com dados de pk retirada de 6 fontes. As opções são 'ds' (DTASelect), 'eb' (EMBOSS), 'ro' (Rodwell), 'so' (Solomon), 'si' (Sillero) e 'wk'(Wikipedia), sendo a última o default.

Os parâmetros 'header' e 'ph' são passados adiante para as funções auxiliares.

A saída é uma matriz contendo o nome da proteína estudada, massa aproximada em daltons, número de cadeias laterais de aminoácidos com carga positiva e negativa, carga da proteína no pH passado como parâmetro, o pl (ponto isoelétrico) da proteína e a fonte dos valores de pk utilizados.

Exemplo de uso:

```
source("protein.diagnostics.r") protein.diagnostics("luciferase", header=T, db = "so", ph = 3.0)
```

Arquivo da Função

[protein.diagnostics.r luciferase.r](#) OBS.: Para usar o arquivo "luciferase", recomendo apagar o ".r" e deixar sem extensão. Em tese, o programa deveria ser capaz de ler o arquivo independentemente do formato, mas a saída tem formato ótimo quando o arquivo de entrada não tem extensão. A função principal é a que tem o mesmo nome do arquivo; as demais funções são auxiliares que eu criei para o

uso da principal, mas que funcionam independentemente e fazem o que suas help pages descrevem.

1)

Em bioinformática, FASTA é um formato baseado em texto para representar sequências de nucleotídeos ou aminoácidos através de um código de letras (resultando em um string de caracteres). Comentários podem preceder as sequências.

2)

Ponto isoelétrico, o pH no qual uma proteína está com carga neutra

3)

Constante de dissociação ácida de cada grupo ionizável, e.g., a cadeia lateral de um aminoácido carregado

From:

<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2011:alunos:trabalho_final:carlos_vido:start



Last update: **2020/08/12 06:04**